Multi-Agent Systems

Introduction

Olivier Boissier
Olivier.Boissier@emse.fr

Plan

1. Definitions
2. Action Domains
3. Positioning
4. Multi-Agent Engineering
5. Perspectives …

Multi-Agent Systems (MAS): set of agents, that interact with each other, situated in a common environment, eventually, building or participating to, an organisation.

Agent (in a Multi-Agent World)

Agent: physical or software, autonomous entity that is pro-active, reactive, social, able to take part to an organised activity, in order to achieve its goals, by interacting with other agents and users.

Definitions

Multi-Agent System (MAS): set of agents, that interact with each other, situated in a common environment, eventually, building or participating to, an organisation.
**Autonomous Agent (in a Multi-Agent World) Definitions**

- An agent $X$ is autonomous with respect to $Y$ for $O$ in situation $S$
  - $Y$ can be a user, another agent, a group of agents, an organisation, ...
  - $O$ can be a goal, a plan, an action, a resource, a norm, a role, ...
- It means that:
  - Agent $X$ can decide locally of the adoption of $O$ in situation $S$
  - And $Y$ has no certainty that $X$ is going to adopt $O$ in situation $S$
- $\Rightarrow$ Loose coupling between agents

**Multi-Agent Systems Principles Definitions**

- The **Agent perspective** (micro perspective)
  - Reactive & Pro-Active entities
  - **Autonomy**: agents may exhibit activities that are not the one expected by the other agents in the system
  - **Delegation**: agents may receive some control over their activities
- The **Multi-Agent System perspective** (macro perspective)
  - **Distribution** of knowledge, of resources, of reasoning/decision capabilities
  - **Decentralization** (loose coupling) of control, authority
  - Agreement technologies, Coordination models and mechanisms to install coordination between the autonomous agents
  - Emergent / Social order / Normative functioning

**Plan**

1. Definitions
2. **Action Domains**
3. Positioning
4. Multi-Agent Engineering
5. Perspectives …

**MAS Action domains**

- **Socio-technical Systems**
  - Integration of software applications, with humans, organizations and the physical world
  - Making them interoperate, interact, cooperate in a flexible and consistent manner with each other
- **Problem Solving**
  - Modeling and solving problems by cooperation between local solvers
  - Installing top-down and/or bottom-up (emergent) solving process
- **Simulation**
  - Modeling and reproducing complex phenomena of interacting entities in the real world in order to understand or to explain their behavior
Socio-Technical Systems (1)

- Industries, services, IT applications are getting global
  - Placed at the centre of multiple networks
  - Developing Knowledge intensive processes
  - Based on large scale underlying IT platforms such as Internet, Web, Internet of Things

- Industries, services, IT applications are situated in an ever-evolving environment
  - Requiring efficient collaboration processes
  - While keeping flexibility and agility

- Users are more and more at the centre of the cooperation and collaboration taking place in these socio-technical systems

Socio-Technical Systems (2)

- Properties of the targeted applications:
  - Absence of monolithic vision
  - Incremental development, by different teams
  - Multi-* (sites, expertise, domains, points of view, decisions, goals, motivations, …)
  - Continuous execution and adaptation
  - User-Centred

- Main requirements:
  - Openness, permeability, scalability in size or structure
  - Distribution, no central control, control and interaction are local
  - Autonomous Interacting entities loosely coupled with others or applications
  - Knowledge Intensive processing and sharing
  - Users may delegate their decisions to the application

Example (1/3)

Service Personnalisation

Tonight’s Suggested Viewing:
7pm World News Headlines
7:15 Personal Neversound
7:30 Selected highlights of today’s golf
7:50-8:00 Intemission (Video call - it’s your brother’s birthday)
8:00-10:00 Film choice Jurassic Park (VR) OR Cyberspace 2
(Source CLIMATE Industrial Workshop 26/4/99)

Example (1/3)

Tonight’s Suggested Viewing:
7pm World News Headlines
7:15 Personal Neversound
7:30 Selected highlights of today’s golf
7:50-8:00 Intemission (Video call - it’s your brother’s birthday)
8:00-10:00 Film choice Jurassic Park (VR) OR Cyberspace 2
(Source CLIMATE Industrial Workshop 26/4/99)
### Example (2/3)

**User-Centred Service Coordination**

- (1) Emergency Assistance
- (2) Telemonitoring & e-Inclusion

**Source:** CASCAM FP6-IST-2

---

### Example (2/3)

**Service Composition**

- Semantic Services Description
- Secured Execution and monitoring of services

**Source:** CASCAM FP6-IST-2

---

### Example (3/3)

**Adaptation & optimisation**

- Planification, coordination, optimisation along a top-down approach:
  - Responsibility Delegation
  - Communication between the nodes
  - Real time detection & reaction to changes
  - Adaptation to changes & continuous optimisation

**Source:** Whitestein Agent Technology Conference 2004

---

### MAS Action domains

- **Socio-technical Systems**
  - Integration of software applications, with humans, organizations and the physical world
  - Making them interoperate, interact, cooperate in a flexible and consistent manner with each other

- **Problem Solving**
  - Modeling and solving problems by cooperation between local solvers
  - Installing top-down and/or bottom-up (emergent) solving process

- **Simulation**
  - Modeling and reproducing complex phenomena of interacting entities in the real world in order to understand or to explain their behavior

---
Problem Solving

- Properties of the targeted applications:
  - Absence of global strategies, of global solving method
  - Interaction between local strategies, between local solving methods
  - Solution is the result of the interaction between local processes (points of view, decisions, goals, motivations, …)
  - Continuous functioning and evolution

- Main requirements:
  - Decentralisation, local control, interactions
  - Openness, permeability, scalability in size or structure
  - Shared and dynamic environment
  - Emergence of the solution

Example (1/2)

Example (2/2)

Design of Complex Systems

- Multi-Disciplinary Simulation & optimisation (ID4CS)
- Design of complex system:
  - Multi-level, Multi-disciplinary
  - Multi-methods
  - Multi-objectives, Multi-attributes
  - Uncertainty
- Cooperation methods between optimisation technics,
- Management of uncertainty
- Multi-* problem solving
- Emergence
MAS Action domains

- Socio-technical Systems
  - Integration of software applications, with humans, organizations and the physical world
  - Making them interoperate, interact, cooperate in a flexible and consistent manner with each other

- Problem Solving
  - Modeling and solving problems by cooperation between local solvers
  - Installing top-down and/or bottom-up (emergent) solving process

- Simulation
  - Modeling and reproducing complex phenomena of interacting entities in the real world in order to understand or to explain their behavior

Example (1/2)

In order to:

Understand, Explain, Discover, …, Help,

Example (2/2)

http://www.massivesoftware.com/

The Return of the King (2003)
The Two Towers (2002)
The Fellowship of the Ring (2001)
Ratatouille (2007)
…, Entertainment

Conversational Zeno Robot
http://hansonrobotics.com/
Plan

1. Contexte
2. Definitions
3. Action Domains
4. Positioning
5. Multi-Agent Engineering
6. Perspectives …

History – Major Steps

- **1980**: Agents in the Artificial Intelligence (AI) area
  - From AI to *Distributed* AI …
  - … to Multi-Agent Systems

- **1990**: Agents are invading other domains
  - Personal Assistants, avatars,
  - Mobile Agents,
  - Reactive Agents, …

- **1995**: Agents spread in other domains, Application domains are enlarging
  - Artificial Life, Economic Agents, …,
  - … Web, Ambient Computing, …

History – Evolutions

- **1973 - 1980**:
  - Hearsay II (1973): blackboard architecture for speech recognition
  - Actor Languages (1973): messages as control structures
  - Beings (1975), Society of Minds (1978)

- **1980 - 1990**:
  - Contract Net (1980): hierarchical decentralized control
  - DVMT (1984): Distributed Interpretation
  - Subsumption architecture (1986): Reactive Robots

- **1990 - …**:
  - Self-organisation, emergence, Interactions, organisations, reputation, trust, Agent Oriented Software Engineering, …
  - In 1995, first international conference ICMAS,
  - since 2002, Autonomous Agents + MAS -> AAMAS

Inter-Disciplinary Domain…

- **Direct Links with**:
  - Programming, Objects…
  - Artificial Intelligence,
  - Distributed Systems, Parallelism,

- **But also**:
  - Complex System (physics, …, ethology, ecology, …)
  - Artificial Life, Neural networks, …
  - Social Psychology, Sociology, Activity Theory, Economy, …
Direct Inheritance

- Object Oriented Programming:
  - Encapsulation, modularity: an object encapsulate data and methods that manage them (ex: C++, Java, Smalltalk),
  - Distribution: Distributed objects, CORBA, DCOM
- → Actor Languages Development

- Artificial Intelligence:
  - Symbolic Reasoning Models (Expert systems, Knowledge Representation), logic, ...
  - distribution: Blackboard Architectures

- Distributed Systems

Multi-Agent vs Objects

- An agent, as an object, encapsulate a state and behaviors
  BUT:
  - An agent encapsulate its control over its behaviors; an object has only control over its state
  - Interactions among agents have a broader scope than the method calls between objects. Interactions consist in goals, plans, actions, hypothesis exchanges
  - An agent may have different control cycles (data-directed, goal-directed, interaction-directed, ...)
  - A MAS has several control flows. An Object system has, a priori, only one control flow.

Multi-Agent vs Artificial Intelligence (1)

- Mono-agent of Artificial Intelligence is rejected
  ➤ Knowledge, Goals, Actions gain a social dimension

Multi-Agent vs Artificial Intelligence (2)

- Ex. dependence networks
Social Knowledge

**Ag3 external description:**
- **goals:** on(A,B),
- **actions:** clear,
- **resources:** A,
- **plans:** on(A,B):=clear(C), put_on(A,B)

**Ag1 external description:**
- **goals:** on(C,Table),
- **actions:** put_on,
- **resources:** B
- **plans:** on(C,Table):=clear(C)

...
Plan

1. Definitions
2. Action Domains
3. Positioning
4. **Multi-Agent Engineering**
5. Perspectives …

Multi-Agent Engineering

- Developing multi-agent applications is often a difficult task
  - implementation, distribution, communications, …

- There exists
  - Multiple technologies focused on particular points of a MAS
  - Multiple agent programming languages, dedicated or general purpose based on existing programming languages
  - Multiple multi-agent programming platforms, involving often a specific agent architecture, proposing or not, first order abstractions for the environment, the organisation, the interaction.
  - Multiple standards
  - Multiple software engineering methods for the analysis and design of MAS.

  → Multiple languages, platforms, methods are available …
  → But often limited to a very focused set of applicative domains.
  
  → Which one to choose? How to choose? How to compare?

Multi-Agent Technologies

- Agent Architectures and Theories
- Coalition formation mechanisms
- Multi-Agent Planning
- Agent Communication Languages, Interaction Protocols
- Auction mechanisms
- Negotiation strategies and mechanisms, Argumentation
- Electronic Institutions, Organisations, Norms
- Reputation, Trust
- Mono & multi-agent Learning
- Self-organisation, emergence, …

Declarative Approach

- CLAIM (Computational Language for Autonomous Intelligent and Mobile Agents)
  - Cognitive Agent Programming Language
  - Belonging to Himalaya Framework (Hierarchical Intelligent Mobile Agents for building Large-scale and Adaptive systems based on Ambients)
  - Based on process algebra in order to represent concurrency and agent mobility
  - Based on SyMPA platform implemented in JAVA respecting the MASIF standard

- FLUX :
  - Cognitive Agent Programming Language
  - Fluent Calculus implementation (Action representation formalism)
  - http://www.fluxagent.org
Imperative Approach

• JACK Agent Language (JAL)
  - Developed by Agent Oriented Software
  - Based on PRS, BDI model (similar to hybrid languages such as Jason, 3APL, Jadex)
  - JAL is an extension of Java allowing to create plans, beliefs base, ...
  - Possibility to use team of agents, organisation of agents

  • http://www.agent-software.com

Languages

Hybrid Approach

• 3APL (An Abstract Agent Programming Language « triple-a-p-l »)
  - Programming language for the development of cognitive agents:
    - By defining structures for beliefs, goals, plans, actions (internal, external or communication) and reasoning rules (modification of plan bases),
    - By reasoning methods to generate plans, revise plans, to achieve goals
    - Integration of Prolog and Java
    - http://www.cs.uu.nl/3apl
  - Jason : extended version of AgentSpeak(L) interpreter, agent oriented programming language based on logic. Introduced by Rao.
  - Communication between agents based on Speech-act (beliefs and goals annotated by the information sources)
  - Plans annotations
  - Functions for selecting, for trust computation,
  - Functions and agent architecture may be adapted (perception, belief-revision, inter-agent communication, acting)
  - Integration by the user of existing code by the way of internal
    - Implemented in java, bound to the organisational language MOISE, interfaced with the CARTAGO platform

Languages

Existing Platforms

• Platforms
  - FIPA compliant
    - FIPA-OS (http://sourceforge.net/projects/fipa-os/)
    - Jade/LEAP (http://jade.tilab.com/)
  - Others :
    - SACI Simple Agent Communication Infrastructure (http://www.lti.pcs.usp.br/saci/)

• Developing Environments
  - Madkit (www.madkit.org)
  - JADEX, BDI agent model based on JADE (http://sourceforge.net/projects/jadex)
  - AgentBuilder based on Agent Oriented Program (AOP) (http://www.agentbuilder.com/)
  - AgentTool (http://macr.cis.ksu.edu/projects/agentTool/agentool.htm)
  - ADELFE (http://www.irit.fr/adelfe/)

  • Have a look at Software Products for MAS, AgentLink, June 2002

Platforms

Standards

• Knowledge Sharing Effort
  - The DARPA Knowledge Sharing Effort

• MASIF - OMG (Object Management Group) : OMG effort to standardize mobile agents - middleware services and internal middleware interfaces
  - www.omg.org

• IEEE Computer Society FIPA Standards Committee
  - (Foundation for Intelligent Physical Agents)
  - www.fipa.org

Standards
FIPA Platform

- Application
- Agent
- ACL
- Agent Management System (AMS)
- Directory Facilitator (DF)
- Message Transport Service

HTTP
IIOP
SMTP
etc.

ACL = Agent Communication Language

JADE (Java Agent DEvelopment Framework)

- Middleware for developing agent-based P2P application
  - On fixed platforms, smart phones, ...
- Two main products:
  - Agent Platform compliant to FIPA specifications
  - API to develop agents in Java
- Open Source Project, LGPL License
- Controlled by Telecom Italia Lab, who owns the project
- Result of the joint effort of multiple actors belonging to the JADE Board (founded in 2003) which missions concern the promotion, the governance and implementation of the changes of the JADE platform
  - Project portal: http://jade.tilab.com

Standards & Multi-Agent Systems

- Ontologies: DAML, OIL, OWL, ...
  - http://www.daml.org
  - http://www.ontoknowledge.org/oil/
  - http://www.w3.org/
- Other standards (De Facto)
  - Jini (www.sun.com/jini),
  - UPnP (www.upnp.org),
  - UDDI (www.uddi.org),
  - Salutation (www.salutation.org)
  - mobility: Aglets (www.trl.ibm.com/aglets/)
  - Web Services (http://www.w3.org/)
  - ...

Multi-Agent Methodologies

The engineering of Multi-Agent Systems needs to take into account two levels:

- Multi-Agent System level (System-Centred)
  - Number of agents, Agent Heterogeneity?
  - What is the common medium shared by the agents (Environment)?
  - What are the communication mechanisms between agents?
  - What are the communication languages, the ontologies, the interaction protocols used by the agents?
  - What is the organisation regulating the actions of the agents? How is it established?
  - How do the agents coordinate their actions? How to ensure a consistent behavior?
- Agent level (Agent-Centred)
  - What does an agent represent? What are the kinds of actions encapsulated into an agent?
  - How do the agents represent the environment, the organisation in which they are situated?
  - How do the agents process the interaction with other agents?
  - What is the agent architecture?
Tools supporting methods

- Software Engineering Tools supporting methodologies:
  - MASE AgentTool: macr.cis.ksu.edu/projects/agentTool/agentool.htm
  - ZEUS: sourceforge.net/projects/zeusagent
  - PASSI ToolKit: mozart.csai.unipa.it/passi/ptk.htm
  - INGENIAS: grasia.fdi.ucm.es/ingenias/
  - OPM: www.objectprocess.org

- Different ways to model applications:
  - Agent Oriented Software Engineering
  - Environment Oriented Software Engineering
  - Interaction Oriented Software Engineering
  - Organization Oriented Software Engineering

Plan

1. Definitions
2. Action Domains
3. Positioning
4. Multi-Agent Engineering
5. Perspectives …

To continue …

- General references
  - Les systèmes multi-agents, J. Ferber, InterEditions, 1995
  - Principe et architectures des Systèmes Multi-Agents, J.P. Briot, Y. Demazeau, IC2, Hermès, 2001

- Some standards
  - Knowledge Sharing Effort http://www.cs.umbc.edu/kse/
  - FIPA http://www.fipa.org
  - W3C http://www.w3.org

- Some general adresses
  - AgentLink : http://www.agentlink.org
  - AgentCities : http://www.agentcities.org

Multi-Agent Modeling

- Multi-model:
  - Articulation of different formalisms
- Multi-viewpoints:
  - Extern/intern, system centred/agent centred
  - Multiple views on a shared world
- Multi-levels
  - Via organisations, via the environment (MAS)
- Multi-scales
  - temporal, spatial, …
Scientific Challenges

- Complexity
  - Applications
  - “massive” MAS
  - “open” MAS
  - “inter-Organisationnal” MAS
- “closed” MAS
- Source: AgentLink Roadmap

Applicative Challenges

- Source: AgentLink Roadmap

Domain Overview (1/2)

- International Conferences
  - International Conference on Multi-Agent System (ICMAS) de 1995 à 2000,
- French Conferences
  - Journées Francophones SMA (http://www.cerv.fr/jfsma08/)
  - Collège SMA de l’AFIA (http://sma.lip6.fr/)
- European Projects
  - AgentLink (réseau d’excellence www.agentlink.org), Roadmap (www.agentlink.org/roadmap)
- Some “Success Stories”
  - BRAHMS (agentsolutions http://agentsolutions.com/home.htm) @ NASA Ames Research Center
  - Living Systems (Whitestein technologies http://www.whitestein.com) @ ABX Logistics
  - eSTAR (http://www.estar.org.uk/) intelligent robotic telescope network
  - CalicoJack (http://www.calicojack.co.uk/)

Domain Overview (2/2)

- Standards
  - FIPA (Foundation for Intelligent Physical Agents) (http://www.fipa.org/)
- Competitions
  - http://www.robocup.org/
  - http://www.rescuesystem.org/robocuprescue/
  - TAC http://tac.eecs.umich.edu/association.html
  - http://www.lips.utexas.edu/art-testbed/
Bibliography


Journals
- Autonomous Agents and Multi-Agent Systems
- Artificial Intelligence
- Knowledge Engineering Review
- International Journal of Agent-Oriented Software Engineering (IJAOSE)
- Web Intelligence and Agent Systems
  - An International Journal

News
- Agent List
  - http://www.cs.umbc.edu/agentslist/
- Distributed Artificial Intelligence List
  - http://www.fipa.org/repository/aclspecs.html
- French list
  - sma@loria.fr
- French list
  - http://www.fipa.org/repository/aclspecs.html

References
**Bibliography**


**Bibliography**